

REMARKS

I. Introduction

In response to the Office Action dated June 2, 2004, claim 7 has been cancelled. Claims 1, 6, 8, 17, and 20 have been amended. Claims 1-6, and 8-20 remain in the application. Re-examination and re-consideration of the application is requested.

II. Prior Art Rejections

A. The Office Action Rejections

In paragraphs (2)-(3) of the Office Action, claims 1-20 were rejected under 35 U.S.C. §103(a) as being unpatentable over Ardoin, U.S. Patent No. 6,052,691 (Ardoin).

Applicants' attorney respectfully traverses these rejections.

B. The Applicants' Independent Claims

Independent claims 1, 11, and 20 are generally directed to providing access to application data items of data objects. Specifically, an extension object is associated with a data object (that contains the data items). The extension object receives a request related to one of the data items. In this regard, the request merely refers to the data object that is associated with the extension object. The extension object fulfills the request by creating a data provider object that is configured to provide access to internal data of the associated data object (i.e., the data provider provides access to the data items). Additionally, if the request concerns at least one other data object, the extension object forwards the request to additional extension objects that are associated with the other data objects.

In view of the above, the claims set forth three specific types of objects: data objects, extension objects, and data provider objects. The data objects contain internal data referred to as data items. Each extension object is associated with a particular data object. The extension object is configured to create a data provider object for the associated data object. The data provider object provides access to the internal information of the data object (i.e., the data item). None of the cited references teach such claim elements.

C. The Ardoin Reference

Ardoin provides a method for maintaining relationships between entities in a computer system, each entity having a plurality nodes, includes the steps of: modifying one of the plurality of nodes; searching for a plurality of dependent nodes from the plurality of nodes coupled to the one node; ordering the plurality of dependent nodes into an order; and evaluating the plurality of dependent nodes in the order.

D. The Applicants' Invention is Patentable Over the References

The Office Action, asserts the following with respect to independent claim 1:

As to claim 1, ARDOIN teaches a method for providing access to application data items (application data) of an application program (CAD/CAM application) (col. 2, lines 27-66; col. 5, lines 3-4), the application data items being contained in a plurality of interconnected data objects (COM objects) processed (via a modification) by the application program (CAD/ CAM application) (col. 5, line 27 – col. 6, line 39), the method comprising the steps of: receiving a request (modification of a node) related to at least one of the application data items (nodes), the request referring to a data object of the plurality of data objects (referenced node, i.e. delete this node, etc.), fulfilling the request with respect to the data object referred to by the request (performing the modification), and if the request concerns at least one other data object of the plurality of data objects (dependent nodes), forwarding the request to the at least one other data object for further processing of the request (change the dependent node order) (col. 73, line 58 – col. 74, line 35; col. 9, lines 30-58; col. 8, lines 15-64). It would be obvious to one skilled in the art that the CAD program accesses the CAD/CAM application data objects / nodes since the application data are COM objects to the CAD.

As indicated above, Applicants have amended the independent claims to incorporate some of the limitations of dependent claims 6 and 7. Specifically, the claims now recite extension objects and data provider objects. Claims 6 and 7 were rejected as follows:

As to claims 6 and 7, ARDOIN teaches the request is a request for initialization of the access (copy request) providing functionality processed by an extension object (dependent node) associated with the data object (node), and the step of fulfilling the request for a data object (node) includes the creation of a data provider (cloning a child node) associated with the data object (node) (col. 9, lines 51-58; col. 58, line 37 – col. 59, line 24; col. 60, line 34 – col. 61, line 5).

Applicants' attorney respectfully disagrees. Firstly, as claimed, the extension object is responsible for creating a separate data provider object that is configured to provide access to internal data of the data object associated with the extension object. Ardoin fails to teach such an extension object or provider object. The Office Action merely refers to a data object as a node and the extension object as a dependent node. However, Ardoin's dependent nodes do not create a third data provider node or object. The cited portions of Ardoin merely describe how a node is copied. There is no description of a data provider that is configured to provide access to internal data. Further, there is no description, implicit or explicit, that a data node that provides such access is created by an extension object that is associated with a data object. Also, Ardoin fails to describe

an extension object (that has performed the above described functionality) that further forward the received request to additional extension objects that are each associated with other data objects.

In view of the above, the cited portion of Ardoin col. 9, lines 51-58 merely describes how to copy a graph containing parent and child nodes. There is no teaching or suggestion that an extension object associated with a data object creates a data provider object. Nor is there any description of a data provider object that provides access to internal data of the data object.

Similarly, col. 58, line 37-col. 59, line 24 merely describes the ability to copy elements. Basically, the cited portion describes that a list/set of objects to be copied is first built and then the set is marked. The marked set is then copied. However, this portion (and the remainder of Ardoin) neither teaches, describes, suggests, or alludes to, implicitly or explicitly: (1) an extension object associated with a data object that is configured to create a data provider object; (2) a data provider object that provides access to internal data of an associated data object; or (3) the forwarding of the request relating to a data object to extension objects that are associated with other data objects.

The last cited portion is col. 60, line 34-col. 61, line 5. This portion of Ardoin merely describes creating a clone graph from an original graph. However, like the other portions of Ardoin (and the remainder of Ardoin), there is no description or suggestion, implicit or explicit, of the extension object, data provider object, creation of a data provider object by an extension object, or the forwarding (by the extension object) of the request to numerous other extension objects.

Thus, Applicants' attorney submits that independent claims 1, 11, and 20 allowable over the references. Further, dependent claims 2-6, 8-10 and 12-19 are submitted to be allowable over the references in the same manner, because they are dependent on independent claims 1, 11, and 20, respectively, and thus contain all the limitations of the independent claims. In addition, dependent claims 2-6, 8-10 and 12-19 recite additional novel elements not shown by the references.

III. Conclusion

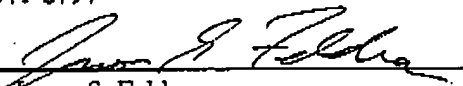
In view of the above, it is submitted that this application is now in good order for allowance and such allowance is respectfully solicited. Should the Examiner believe minor matters still remain that can be resolved in a telephone interview, the Examiner is urged to call Applicants' undersigned attorney.

Respectfully submitted,

GATES & COOPER LLP
Attorneys for Applicant(s)

Howard Hughes Center
6701 Center Drive West, Suite 1050
Los Angeles, California 90045
(310) 641-8797

Date: September 2, 2004

By: 
Name: Jason S. Feldmar
Reg. No.: 39,187

JSF/sjm